UNIVERSITY OF SOUTHERN CALIFORNIA ISI INFORMATION SCIENCES INSTITUTE
4676 Admiralty Way, Marina del Rey, CA 90292

Next Generation Internet (NGI)
Multicast Applications and Architecture (NMAA)

Final Technical Report

Period: 06/28/1999 – 12/31/2002

Sponsored by
Defense Advanced Research Projects Agency (DARPA)
Contracts Management Directorate (CMD)

Under Contract No: MDA972-99-C-0022

APRA Order No: H645-02/03

The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Defense Advanced Research Projects Agency or the U.S. Government.

REPORT DOCUMENTATION PAGE

FORM APPROVED
OMB NO. 0704-0188

1. REPORT DATE (DD-MM-YYYY)
31-1-2003

2. REPORT TYPE
Final Technical Report

3. DATES COVERED (From - To)
28-6-1999 to 31-12-2002

4. TITLE AND SUBTITLE
Next Generation Internet (NGI)
Multicast Applications and Architecture (NMAA)

5a. CONTRACT NUMBER: MDA972-99-C-0022

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)
Colin Perkins

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
USC INFORMATION SCIENCES INSTITUTE
4676 ADMIRALTY WAY
MARINA DEL REY, CA 90292-6695

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Defense Advanced Research Projects Agency
Contracts Management Directorate
3701 N. Fairfax Drive
Arlington, VA 22203-1714

10. SPONSORING/MONITOR'S ACRONYM(S)
DARPA/CMD

11. SPONSORING/MONITORING AGENCY REPORT NUMBER

12. DISTRIBUTION AVAILABILITY STATEMENT
UNCLASSIFIED/UNLIMITED

13. SUPPLEMENTARY NOTES

14. ABSTRACT

The NGI Multicast Applications and Architecture project has developed innovative technologies and standards that greatly increase the quality and scalability of IP-based multicast teleconferencing and real-time motion imagery distribution systems. These scalability enhancements fall into two categories: 1) improved support for large-scale distributed meetings; 2) improved support for distribution of high-definition video. In particular, we have demonstrated a prototype Digital Amphitheatre supporting virtual meetings with hundreds of simultaneous teleconferenced participants, and a prototype system for delivery of gigabit-rate High Definition video over commodity IP networks. These capabilities are significantly beyond those available commercially, and leverage the advanced network infrastructure developed as part of the DARPA Next Generation Internet research program.

15. SUBJECT TERMS
Teleconferencing, HDTV-over-IP, networked multimedia, agent-based scalable virtual environment, Digital Amphitheatre, NGI SuperNet.

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON**
a. REPORT	**b. ABSTRACT**	**c. THIS PAGE**	UNCLASSIFIED	24	Colin Perkins
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED			**19b. TELEPHONE NUMBER** (703) 812-3705

NGI Multicast Applications and Architecture

Report by Colin Perkins on work by Aaron Griggs, Jaroslav Flidr, Ron Riley, Maryann Perez Maher, Michael Craig, Ladan Gharai, Colin Perkins and Allison Mankin

University of Southern California
Information Sciences Institute

1 Introduction

The NGI Multicast Applications and Architecture project has developed innovative technologies and standards that greatly increase the quality and scalability of IP-based multicast teleconferencing and real-time motion imagery distribution systems. These scalability enhancements fall into two categories:

1. improved support for large-scale distributed meetings; and
2. improved support for distribution of high-definition video

In particular, we have demonstrated a prototype Digital Amphitheatre supporting virtual meetings with hundreds of simultaneous teleconferenced participants, and a prototype system for delivery of gigabit-rate High Definition video over commodity IP networks. These capabilities are significantly beyond those available commercially, and leverage the advanced network infrastructure developed as part of the DARPA Next Generation Internet research programme.

This report describes these technologies, and is structured as follows: in section 2 we describe the aims and objectives of our research; section 3 describes the technical problems to be solved in achieving these objectives; section 4 outlines our methodology; sections 5 and 6 describe our results, findings and conclusions; section 7 describes the systems we have built; and section 8 describes the implications for future research.

2 Task Objectives

The objectives of the NMAA project were to demonstrate large scale multicast and high definition teleconferencing, leveraging the vastly increased performance of the Next Generation Internet to provide capabilities significantly beyond those previously envisaged. To demonstrate these capabilities, we chose two applications:

1. The Digital Amphitheatre, being a shared virtual environment where hundreds of people can share an information experience, attend a lecture, or participate in a discussion.
2. High Definition Video Teleconferencing and Motion Imagery, bringing a vastly enhanced sense of presence to networked video teleconferencing, and enabling new applications that require very high fidelity imagery.

Together these two applications stress the limits of existing networks, protocols, and architectures in two axes: the number of simultaneous participants in a teleconference, and the media quality/data rate. We discuss each demonstrator application in turn.

2.1 Digital Amphitheatre

The aim of the Digital Amphitheatre is to create a digital meeting place, an environment where participants in the meeting can feel that they are interacting with each-other; rather than using a complex teleconferencing system. The system mimics an auditorium, with seating for the audience and a panel of speakers, much as one might find in a typical meeting or seminar. To implement this on a flat display, the audience is reflected, so each participant sees a view from the stage showing their presence with the other audience members, but we show the speaker and panellists as if viewed from the audience. Figure 1 illustrates the concept, with a mock-up we used in our early design.

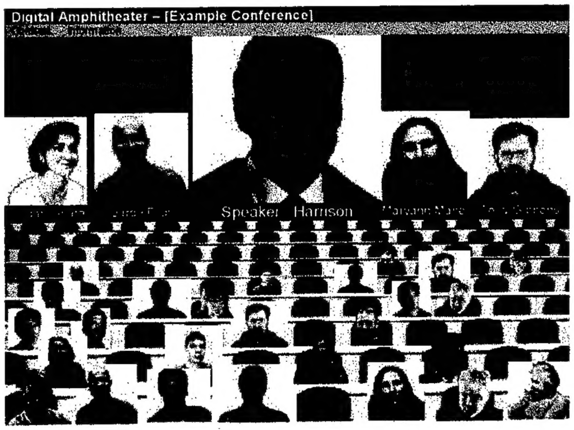


Figure 1: Mock-up of the Digital Amphitheatre

In this mock-up, the images of the participants have been processed to remove their background. Each participant is seated in an amphitheatre seat. The seating follows the rules of perspective, such that seats and participants become smaller as they move towards the back. The use of background substitution and natural seating provides the illusion of presence, and allows a large number of video images to be composited whilst maintaining a visually pleasing aspect.

While the participants are scattered through out the amphitheatre, the speaker appears in the middle of the front row amongst other panel members. The speaker occupies a relatively large video frame (possibly with high frame rate) as do other panellists. Both speaker and panellists have their names written in front of them, as they would in an actual panel session.

There is no moderator in place, therefore it is possible for everyone to talk at the same time, although of course the result would be a difficult to understand jumble of sound. Again, our model is based on real-life conferences, where floor time is dictated by social norms.

The system is capable of supporting several hundred simultaneous interactive users, limited by the available screen real-estate. The benefits of such a system are obvious: large organizations can have regular meetings with all levels of management involved without incurring high travel cost, long distance educational programs can meet as if within a lecture hall while students and lecturers join from geographically disparate locations, or it could be used for political and other debates. Our Digital Amphitheatre will demonstrate the key components of such an interface, illustrating the feasibility of large scale distributed meetings.

2.2 High Definition Video Teleconferencing and Motion Imagery

One area where systems for IP-based video teleconferencing and motion imagery have been lacking is in support for high definition content. Desktop teleconferencing systems typically provide windows comparable in size to a postage-stamp, while stand-alone units provide physically larger, but still low resolution images, inferior even to the quality provided by a standard television signal. Factors such as these significantly limit the applicability of these systems.

Using the advances in network capacity provided by the Next Generation Internet, and new availability of very high performance commercial motion imagery systems, we proposed to develop a demonstrator for the next generation of high definition video teleconferencing and motion imagery systems. This system aimed to support current and future applications of networked motion imagery, including:

1. teleconferencing for high-level meetings, where it is required that you can see the expression and mannerisms of the remote participant, and have a realistic sense of presence;
2. distribution of surveillance imagery, and other real-time high-resolution content; and
3. remote operation of systems where high definition imagery is necessary for accurate control.

To achieve these aims, we will vastly improve the image resolution, frame rate and colour fidelity available in IP-based video teleconferencing and motion imagery systems. We aimed to combine the highest quality video commercial motion imagery format with state of the art computer and networking technology, to provide a demonstrator video teleconferencing system that exceeds the image resolution available in commercial products by at least an order of magnitude, and provides double the maximum framerate.

3 Technical Problems

Our intention was to scale teleconferencing systems to support large-scale distributed meetings, with hundreds of simultaneous transmissions, and to support very high quality teleconferencing and motion imagery, at rates many times those of existing systems. These demonstrators were significantly beyond the state-of-the-art at the project inception, and illustrated a number of technical problems to be solved before they become reality.

Considering the Digital Amphitheatre demonstrator, we considered solutions to the following issues:

- Large scale distributed meetings are restricted in scope due to the limitations of end-system processing ability. It is clear that the ability of a single host to receive data cannot be scaled to match the ability of hundreds of hosts to send data to it; therefore we sought a solution by which video processing can be distributed within the network to reduce the load on any single receiver.
- Given the existence of distributed processing, it is necessary to include a location service by which the closest processing agent can be located. This location service must be integrated into the session initiation infrastructure, such that joining a distributed meeting becomes a straightforward matter.
- It is clear that the usability of large-scale environments for distributed meetings is limited, due to the cluttered user interface they present. We sought a solution to this problem that could eliminate the clutter and provide increased sense of presence, whilst still providing the needed features.

In constructing our demonstrator for high definition motion imagery and video teleconferencing, we considered the following issues:

- While the protocol standards for video teleconferencing and real-time media distribution are mature, they have not been used to convey content of the quality and data-rate we envisage. It will be necessary to investigate the following:
  - It is unclear if timers, counters and other components of the Real-time Transport Protocol, RTP, have been designed to operate at the rates we require. In particular, it is unclear if the infrastructure for quality-of-service feedback, jitter compensation and error resilience is sufficient. It is necessary to study the protocol to confirm that it supports our needs, and if necessary recommend enhancements to the standards committee.
  - The RTP framework supports the notion of *payload formats* that adapt it to particular media formats. There is no standard payload format for high definition motion imagery, and hence it is necessary to develop such a format and integrate it into the RTP framework.
- Given the data rate of high definition video, it is unclear if existing hardware and operating systems can perform video capture and/or playback whilst network I/O is ongoing. Of particular concern is the performance of the PCI bus and operating system interrupt handlers. We expect to see considerable bus contention and interrupt processing overhead, that will – at best – require careful system tuning, and at worst may restrict system performance.
- Implementations of the RTP protocol and system UDP/IP network stack are not typically optimised for high-performance. There has been considerable work on improving HTTP and TCP/IP performance, but this does not directly apply to the needs of real-time applications. We expect it will be necessary to conduct significant protocol and application optimisation work, to achieve the necessary performance.
- It is not clear that the Next Generation Internet can support the data rates we requires in a robust and loss free manner, and without significantly impacting the packet timing. We expect that it will be necessary to develop mechanisms for error correction and/or concealment, and timing recovery, to make our system robust to the vagrancies of the network. In addition, we expect that it will be necessary to conduct network performance monitoring and evaluation, to determine the causes and effects of various conditions.

Together, we expect our demonstrator applications to significantly probe the limits of our knowledge and system performance in both the axes scalability.

4 General Methodology

We intend to solve these technical problems using a practical approach, leveraging standard protocols and hardware, combined to make innovative architectures that demonstrate the novel capabilities we have proposed. Our approach will be multi-faceted, along the following directions:

- A comprehensive literature review will be conducted, focussing on both research work and the capabilities of existing protocol standards and systems. The aim is to identify the standard components that can be leveraged, the research results that can be applied, and the areas where further research and development is needed.

- We will design innovative demonstrator systems, based on the results of our literature survey. Where necessary, this will involve research and development in the areas of network protocols, operating systems, system performance evaluation and tuning, and network performance and tuning. Results will be published in the open literature, as they become available.
- We will develop laboratory prototypes, based on commercial-off-the-shelf hardware, that implement the demonstrator applications. These will be tested to evaluate their performance, both using local test bed facilities, and across the NGI SuperNet test bed infrastructure. The software components of our prototype systems will be made available to the community, along with details of the necessary hardware and network infrastructure required to evaluate their use. It is expected that this open source development model will encourage use and experimentation by other parties, enhancing the value of the research.
- Results will be fed back to DARPA and to the community, through our participation in the standard process. This will help to ensure that future generations of commercial systems will support the needs of DARPA, and improve interoperability between implementations.

Our focus is on the practical application of Next Generation Internet technology, demonstrating novel capabilities and architectures, and on the advancement of the industry through the development of next generation standards and systems.

5 Technical Results

5.1 Digital Amphitheatre

Video teleconferencing among small groups of people is now quite common, and is supported by a number of commercial and open-source tools. However large structured meetings, on the scale that we envision for the Digital Amphitheatre, have not yet been tried. There are a number of reasons for this: processing such a large number of video streams presents a formidable challenge, both in the network and for the end-user application, and display technology is often a limiting factor. Processing hundreds of video streams can easily overwhelm most workstations, in terms of bus access, interrupt processing, context switching, packet handling and de-multiplexing, decoding, display processing and rendering.

Many of the current teleconferencing tools, especially the research oriented ones such as the popular "Mbone conferencing" toolset have been designed with scaling properties in mind. However, their focus has been mainly on attaining scaling via multicast, and thereby reducing network load. This approach does not address the problem of the end-system bottleneck, and in fact it aggravates it. End-users can generate video content in parallel, this content moves through the network, but once received at its destination, must be processed by an inherently serial system. As all the video flows must be instantaneously reconstructed, decompressed and rendered, thereby creating a performance bottleneck in the end-system.

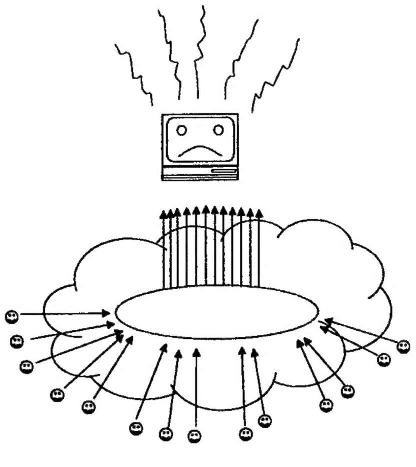


Figure 2: Parallel Generation of Content Overwhelms End System

Given that the processing limitations of end-systems are the main bottleneck and deterrent to very large scale video conferencing, what are the possible solutions? Our experience shows that the simple brute force technique of "faster end-systems" is not a viable solution, as even the fastest available workstations cannot keep up with hundreds of video streams.

The implication is that we must distribute the processing using application level multicast and active agents, leveraging the increased communication ability rather than drinking from the fire hose of the full set of input streams. Parts of processing must be pushed into the network infrastructure, offloading functions from the end-system to agents within the network. The question remains as to how much and which parts of the process can be off-loaded from the end-system, and exactly what are the tradeoffs involved.

5.1.1 System Architecture

To support a large number of video streams in the digital amphitheatre we adopted an agent-based approach, distributing the processing required to build the user interface throughout the network. There are several parts to the system: background substitution at the transmitter, spatial tiling agents within the network, and user interface composition at the receiver.

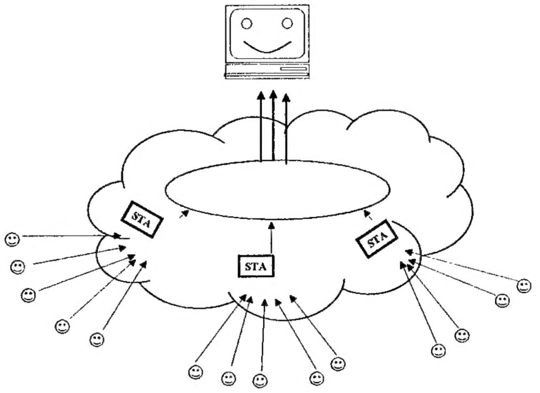


Figure 3: Spatial Tiling Offload Processing into the Network

Each transmitter performs the background substitution algorithm on their own video, replacing the actual background with a synthetic image supplied during session initiation. Each audience member participates by unicasting video to the closest tiling agent. The agent, in turn, tiles together all the video streams it receives, and sends the resulting stream to a multicast group. All participants join this group, receiving and displaying the combined audience video. The panellists and speaker send directly to the multicast group, thus circumventing the tiling agents.

The receivers compose the tiled audience segments, speaker and panellists into a single display. Audio is received directly via a single multicast group, since it is expected that the audio rate will be low (silence is suppressed, so there is typically only a single active audio sender).

In addition to distributed processing based on media agents, control protocols are needed to announce and setup the session, enabling the participants to find the tiling agents and each other. The session can be announced using SAP, SIP, a web page or even email. The announced session has a single piece of information within it: an anycast address, which should be contacted via SIP to obtain the details needed to join the session.

On sending a SIP request to that anycast address, the routing system will ensure the response comes from the closest member of the anycast group. This will be a SIP server, co-located with a tiling agent, which will respond to this request and return both the multicast group used for the audience, and unicast address of the closest tiling agent. A user can then participate by sending video to either the unicast address or the multicast group, respectively.

This architecture spreads the processing load throughout the network, while maintaining a simple method of joining the session.

5.1.2 Background Substitution

An important motivation in our design of the Digital Amphitheatre was providing a feeling of presence, so that all participants feel as if they are in the same location i.e., an amphitheatre, a classroom or other meeting place. Of course, this necessitates removing differing backgrounds from each participant, and substituting them with a common background of choice, as shown in Figure 4.

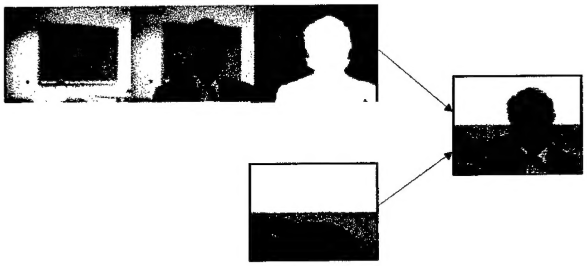


Figure 4: Background Substitution

The background substitution process requires an initial background image to use as a baseline for comparison. Once the camera has been positioned and adjusted for use during the meeting, the participant moves out of the field of view of the camera for a few seconds to allow the software to collect several frames of the background. These images are averaged together to provide a low-noise estimate of the background.

After this brief training period, the participant returns to his seat. The region of the current video image that has changed significantly from the background is then segmented from the rest of the image, allowing the background to be substituted, using a textbook background segmentation algorithm.

A direct comparison between the current and background frames is made difficult by features common to many commodity video cameras including lighting changes, automated exposure, dynamic white balance, and increased noise. For this reason, there is a scaling step in our algorithm: we compare pixels primarily on their colour, but allow the apparent intensity to vary in order to compensate for changes in brightness. The resulting distances are thresholded to produce a binary mask labelling the pixels as foreground or background.

It is also possible that natural backgrounds, such as an office, contain small regions that are difficult to distinguish from the foreground. We apply morphological operators to the mask to compensate for small regions of anomalous colour match: the mask is eroded by a radius of two pixels to remove most of the isolated regions caused by noise in the current frame; the mask is then dilated by roughly twice the erosion radius to fill in voids (an additional erosion step may be performed, depending on the amount of noise in the image); the mask is finally eroded once more such that the total number of erodes and dilates balance to zero to restore the outer boundary of the foreground

We use a low-complexity algorithm, with acceptable performance. Performance suffers when the background is subject to large changes in lighting: a more dynamic approach to updating the stored background image would improve performance. The system also has to be retrained if the background image changes, although fortunately training is a simple process.

5.1.3 Spatial Tiling Agents

Each attendee in the amphitheatre participates by unicasting video to their closest tiling agent, located via an anycast address. The tiling agents combine video streams from several sources to produce a single high-bandwidth tiled stream in the place of the individual, lower bandwidth, video streams. All participants join the group, receiving and displaying the combined audience video. The panel members and speaker send directly to the multicast group, thus avoiding the tiling. Scalability comes because the key limiting factor end-host performance is the per-packet processing overhead, and the tiling process can reduce the number of packets by an order-of-magnitude.

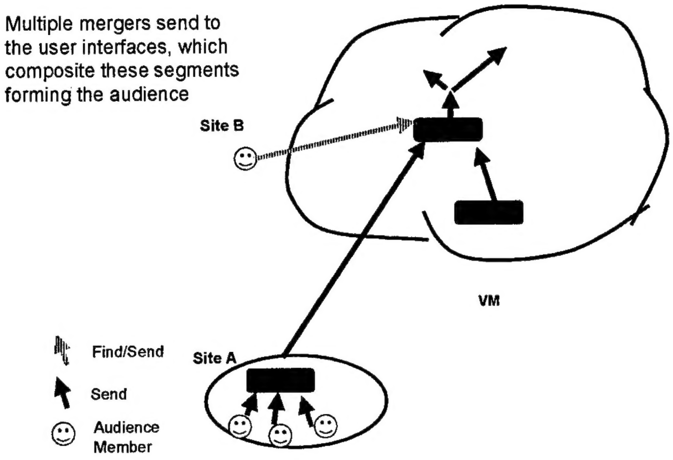

Multiple mergers send to the user interfaces, which composite these segments forming the audience
Site B
VM
Find/Send
Send
Audience Member
Site A

Figure 5: Use of Spatial Tiling Agents

To illustrate the spatial tiling operation, consider the example in Figure 6: three streams of video are spatially tiled and represented as a single frame. The tiled frame consists of the three frames side by side, with each of the frames being completely represented. The meta-data for each of the frames, in this case the frame size and block coordinates, are adjusted accordingly. The resulting frame size is the total frame size of the tiled frame, and block coordinates are transposed to the correct location.

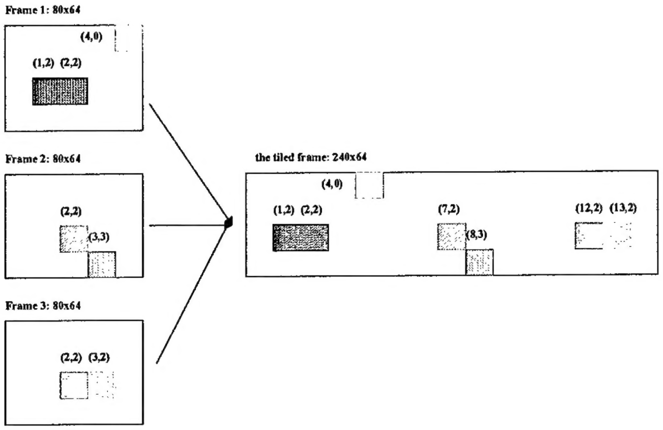


Figure 6: Tiling Several Frames

It is important that spatial tiling does not add additional delay to the video stream. Tiling agents only parse and deconstruct the incoming video streams into smaller building blocks, whilst maintaining their relevant meta-data: no decompression is done in the tiling agent. To maintain independence between incoming and outgoing frame-rates, two sets of buffers are maintained per stream. The tiled frame is constructed at given intervals (determined by the outgoing frame rate) from the output buffers. New frames are copied from the incoming buffer to the output buffers, once they are received in full.

Although, theoretically, it is possible to tile an unlimited number of streams up to the maximum transfer unit (MTU) of the network, we have restricted the tiling to 15 video streams. This restriction allows us to use the built in mixer functionality of RTP/RTCP, since an RTP packet can carry the contributing source identifiers for up to 15 different sources. The input streams can be tiled in any geometry requested: for 15 streams the agent can generate a single row of 15x1, a square of 4x4 (where the last square will be empty), a 5x3 rectangle, or even a single row/column.

In our current implementation, the spatial tiling agents support two video representations: high bandwidth raw video in component YUV form, with conditional replenishment (YUVCR) and H.261 using only intra-frame compression. At the receiver, any YUVCR decoder can receive and display a tiled YUVCR stream. However, for H.261, we have added an H.261 tiled decoder, H.261t, to the video conferencing tool in order to receive and decode a tiled H.261

stream. This was necessary as the tiled H.261 stream no longer complies with the standard H.261 syntax, due to the large frame sizes. Both the standard H.261 syntax and the RTP payload headers have been slightly modified.

The simple structure of YUVCR is well suited to spatial tiling. Each video frame is divided into macro blocks of 16x16 pixels, represented in planar YUV format with a 4:2:0 colour sub-sampling. The conditional replenishment algorithm insures that only updated video macro blocks are transmitted, and provides for some reduction in the data rate, in what is otherwise raw video. Unlike H.261, no meta-data related to the frame is carried in the frames themselves. The size of the video frames and the macro block coordinates are carried in an RTP payload header. For example, Figure 7 displays the RTP payload for frame 2 of our tiling example, where macro blocks (2,2) and (3,3) of the frame have been updated and must be transmitted. The RTP payload contains first the height and width of the video frame (80x64), followed by the coordinates and data of the two updated macro blocks. This structure not only makes for a high degree of flexibility in terms of frame sizes, up to 4096 pixels in each axis (the numbers are actually stored in multiples of 8), it also lends itself very well to spatial tiling.

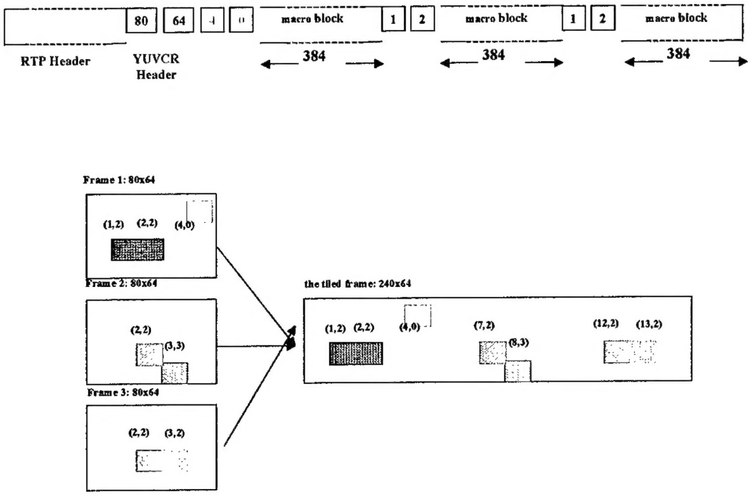


Figure 7: Tiling YUVCR

Tiling YUVCR frames essentially consists of manipulating the size of the video frame in the RTP payload header and the coordinates of each 16x16 macro block such that it reflects the position of the macro block in the new tiled frame. As the information in the RTP payload is sufficient for tiled YUVCR, it is unnecessary to change the RTP payload, therefore the YUVCR decoder need not be altered either. At the receiving side, the YUVCR decoder is unchanged, it renders and displays the tiled frame in the usual manner.

Tiling H.261 frames is, in essence, the same as tiling YUVCR frames, since H.261 also divides each video frame into 16x16 macro-blocks, which are the smallest building blocks that the STAs process. However, tiling H.261 is somewhat complicated by intricate header system used to describe a frame and the use of Huffman encoding. Figure 8 shows the syntax diagram for an H.261 video frame. As is obvious from the diagram, extracting a macro-block requires manipulating non-byte aligned bit values and variable length fields. In this diagram following the solid lines in the macro block layer, produces the headers for intra-frame H.261.

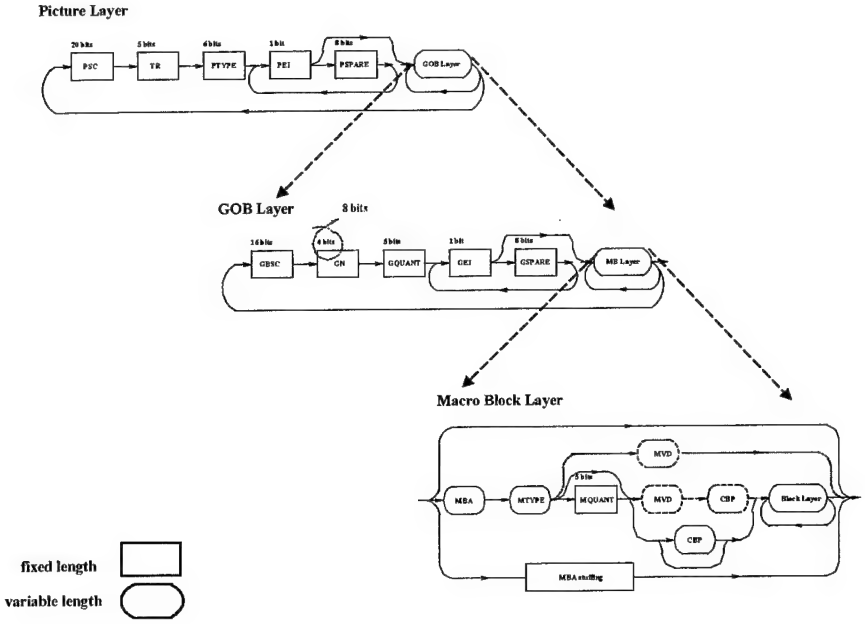


Figure 8: Syntax of an H.261 frame

An H.261 video frame consists of three layers: a picture layer, a group of block (GOB) layer and a macro block (MB) layer. Each GOB is divided into 33 macro-blocks, arranged in a 3x11 matrix. H.261 supports two scanning formats CIF and QCIF. A CIF frame contains 12 GOBs number consecutively from 1 to 12, whereas a QCIF frame contains 3 GOBs numbered 1, 3 and 5.

In the tiled H.261 frame, GOBs are numbered consecutively from 1 to Nx3 + Mx12, where N is the number of QCIF frames and M is the number of CIF frames tiled (currently both N and M cannot be non-zero, tiling of mixed CIF and QCIF frames planned for a future version). Since the standard H.261 GOB header only allocates 4 bits to the GOB Number (GN) field, it is necessary to extend this field to 8 bits in the tiled frame, so as to accommodate up to 15 CIF frames. GOB numbers are hence renumbered within the STAs prior to packetization. No changes are made to macro block headers: each macro block header is copied to the tiled frame as is. The tiled frame is preceded by a single picture header.

Overall, the primary changes made to the tiled H.261 frame, are replacing the individual picture headers by a single picture header and extending the GN field to 8 bits. For a tiled frame of 15, using a single picture header results in 56 bytes of savings (4 bytes per picture header) and the 4 bit increase in the GN field adds 12 bits per QCIF frame, or 22.5 bytes for a tiled frame of 15. All in all, 15 tiled QCIF frames save 33.5 bytes when compared with the non-tiled frames. For CIF frames the additional 4 bits per GOB, adds up to 6 bytes per frame. Therefore when 15 CIF frames are tiled, the tiled frame is 34 bytes larger than the individual non-tiled frames. However, this increase in ameliorated by the reduction in the number of packets and per packet overhead (40 bytes for each IP/UDP/RTP header) once the frame is packetized.

The increase in the size of the GOB number field must also be reflected in the standard RTP payload for H.261, which only allocates 4 bits for the GOBN. We define a new RTP payload header for tiled H.261 where GOBN is extended to 8 bits. To maintain the 4 byte size of the payload header, for intra-frame H.261, we have eliminated the two motion vectors from the payload header (since our implementation does not currently support motion vectors).

Other information needed for by the H.261t decoder is the requested formation of the tiled frame, i.e., is the tiled frame a row of 15 or a block of 3x5? This information is signalled out of band to the STA and the number of tiled frames is extracted from the number of contributing sources. Thereby allowing the H.261t decoder to deduce the height and width of the tiled H.261t frame.

5.1.4 Performance of the Spatial Tiling Algorithm

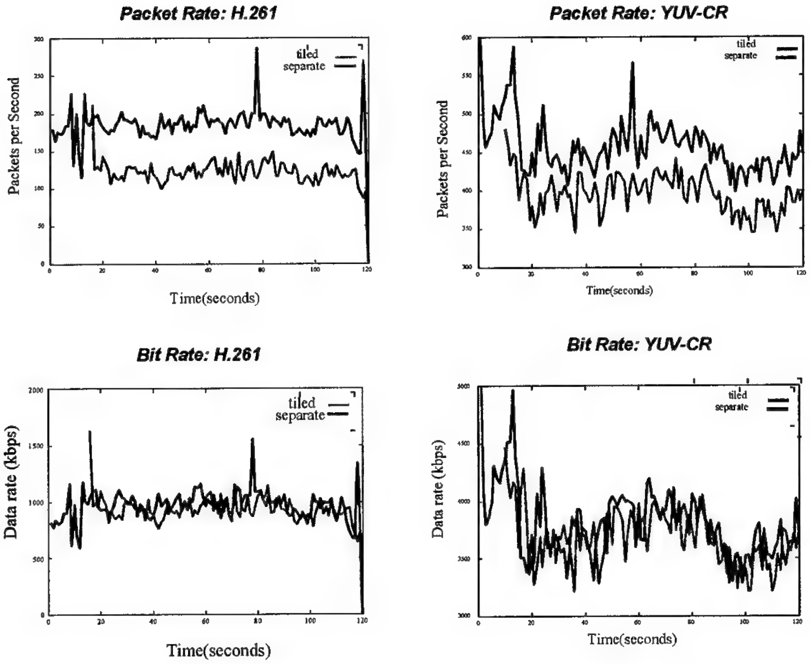
The main goal of our performance measurements was to answer the following question: *What benefits are gained by using spatial tiling agents?* To identify potential performance gains we decided to measure and quantify: (1) bandwidth, in bits per second; (2) the packet rate, per second; and (3) the total number of streams the end system is capable of decoding and rendering (N). We compared the value of these variables in a conferencing session with and without the use of STAs. The test material comprised 15 YUVCR streams and 15 H.261 streams, all recorded at 8fps and 2 minutes long.

The receiving system was what was considered an average user grade system at the time: a 550Mhz Pentium III machine with 256M of memory, running Red Hat Linux 7. The STAs were initially run on a somewhat lower grade system, a 400Mhz Pentium II with 64M memory, running FreeBSD 3.4, but this was found to have insufficient memory to run multiple STAs, although it was sufficient in other ways. A more powerful system, with 512M of memory, was used to host the tiling agents during the tests we report. Work is underway to reduce the memory footprint of the tiling agents, since they are otherwise not very compute intensive and require only a small percentage of CPU time.

In our initial set of trials, we measured the bit rate and packet rate. To do so, first we streamed the 15 test video streams individually to the Digital Amphitheatre prototype. Next, we ran the test video through the spatial tiling agents with the output frame rate set to 8fps. To measure the bit rate and packet rate we instrumented the Digital Amphitheatre system such that it logged these variables, along with other decoding statistics, to a file.

Figure 9 below shows the reduction in packet rate is due to the aggregation of smaller packets. The tiling agents generate a single large frame, therefore there are fewer `half empty' packets in the resulting stream. In the tiled YUVCR stream the packet rate is reduced by approximately 13% and in the H.261 tiled stream, the packet rate is reduced by approximately 35%. The higher reduction in the packet rate of the H.261 tiled stream in our trials is the result of the more flexible nature of H.261 and its compression scheme. H.261 packets, range anywhere in size from 48 bytes to 1024 bytes. A YUVCR packet, on the other hand, can only hold one, two or three macro blocks, which results in packets sizes of 400 bytes, 786 bytes and 1172 bytes. This means there are less options on how to aggregate packets for the YUVCR tiled stream, whilst remaining within the 1500 byte Ethernet MTU, which results in less reduction of the packet rate.

In terms of bandwidth, the tiled YUVCR stream is reduced by an average of 3% and the H.261 stream by 4%. Although bandwidth is reduced over the duration of the test runs, the graphs reveal that this in not the case on a per minute bases, as in some instances the bandwidth of the separate streams appears to be less than the tiled stream. This is in part due to synchronization differences between the separate streams and the tiled stream, and in part due to measurement artefacts resulting from the averaging process. We also note the small reduction in bandwidth is to be expected: in these tests the tiling agents exactly reproduce the input video streams, without any temporal or spatial down sampling. Both the input and output frame rates of the STAs are 8 fps and the tiling agents more or less copy each incoming frame to the outgoing tiled frame. There are some instances where incoming frames are not pushed out by a tiling agent due to synchronization variability between incoming and outgoing frame rates, this can results in a dropped frame. However as our data shows this does not happen often. The existing reduction in bit rate is mainly a reflection of the reduced per-packet overhead due to the tiling process.



Packet Rate: H.261

Packet Rate: YUV-CR

Bit Rate: H.261

Bit Rate: YUV-CR

Figure 9: Change in Packet Rate and Bit Rate due to Spatial Tiling

Finally, we turned our attention to the performance of the end-system, and quantifying the number of video streams that can be supported with the aid of the tiling agents. Our decoder maintains statistics on the number of packets correctly decoded and on packets discarded due to late arrival or lack of rendering time. We used these statistics to measure the maximum number of streams, N, the Digital Amphitheatre could receive without loss, both with and without the help of the tiling agents. This process was conducted by incrementally increasing the number of individual streams until the end-system reached the point of saturation. For the individual H.261 video streams it was found that the system could decode and render up to 55 individual video streams without loss. With this number of streams CPU was at 100% utilization. When receiving tiled H.261, the system could receive 6 tiled streams of 15 and an additional stream of 2 tiles, comprising a total of 97 individual streams, an overall increase of 43% in number of streams. For YUVCR, our end-system was able to receive 42 individual streams without loss, while tiled, the end-system could process 3 fully tiled streams and one tiled stream of 12, a total of 57 individual streams.

These numbers clearly demonstrate the reduction of workload on the end-system due to the spatial tiling process. For our H.261 streams the end-systems is capable of receiving almost twice as many video streams, once the video streams are tiled and for YUVCR the number of streams is increased by about 25%. The greater increase for H.261 is in part a reflection of the greater reduction in packet rate for the H.261 streams. In fact in all our tests H.261 fared better than YUVCR. Although YUVCR requires no processing time, its higher data rates and non-flexible packetization scheme, seem to make it an unsuitable candidate for spatial tiling.

This leads us to conclude that a primary load on end-systems is per packet interrupt processing rather than the computational complexity of the decoding process, and therefore spatial tiling is more amendable to relatively highly compressed video streams where the average packet size is significantly smaller than the network MTU. Having a significant number of half-full packets, gives the STAs more leverage in reducing the overall packet rate.

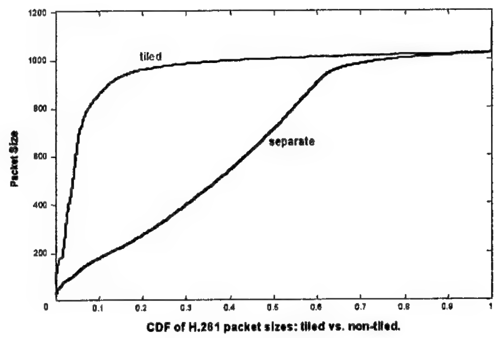


Figure 10: Performance of the Overall System, Showing Effects of Spatial Tiling

5.1.5 Composition of User Interface

A key aspect of the Digital Amphitheatre is its innovative user interface. The prototype interface has been implemented as an addition to the *vic* video conferencing application from Lawrence Berkeley National Laboratory, integrated with the UCL Robust-Audio Tool using a message bus infrastructure we have developed.

The user interface of *vic* has been augmented with an additional mode, which displays the speaker, four panellists, and a number of audience segments (in our present implementation, six segments are used) in a full-screen configuration. Due to the use of spatial tiling agents, it is only necessary to display a small number of video streams for the audience: each stream contains a 5x3 block of 80x64 pixel frames. The result is an interface that displays hundreds of video clips, while only receiving a small number of RTP streams (our present implementation receivers 11 streams in total, yet displays 95 video sources). Figure 12 shows the combination of streams via the spatial tiling agents, and their composition in the user interface.

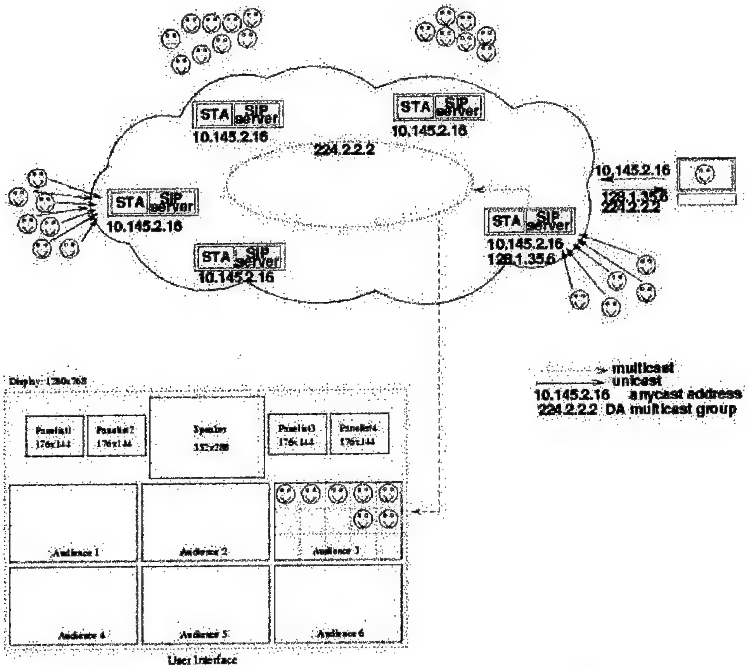


Figure 11: Composition of the user interface

Due to the use of background substitution, the interface is clean and simple (Figure 12). It displays no information about each source by default: a tool tip popup is used to highlight participant names and other information (the mixer functionality included in RTP allows for this to be conveyed along with a tiled video stream).

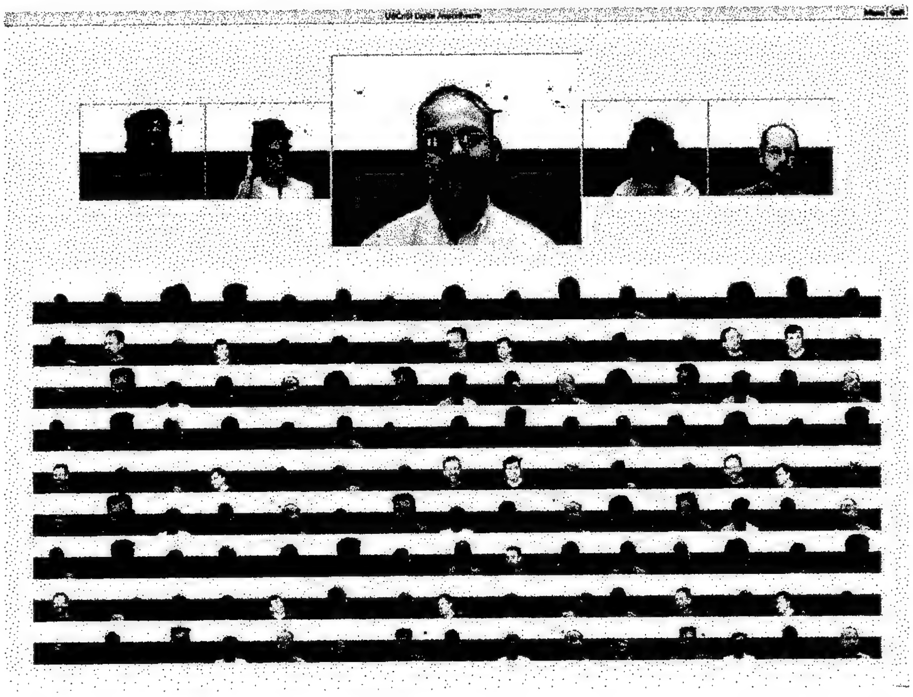


Figure 12: Screenshot of prototype user interface, showing 140 simultaneous video streams

5.2 High Definition Video Teleconferencing and Motion Imagery

In support of our goal of bringing high definition video teleconferencing and real-time motion imagery to the Internet environment, we sought to combine the highest quality motion imagery format available with state of the art computer and networking technology. The result is a demonstrator teleconferencing system that exceeds the image resolution available in commercial products by at least an order of magnitude, and provides double their maximum frame-rate.

To achieve this, we built our prototype leveraging High Definition Television (HDTV) technology and high-performance PC hardware, linked through gigabit networks such as the NGI SuperNet. In the following, we outline the components of this prototype – HDTV and standards for IP-based multimedia – and then describe the system architecture, design and performance.

5.2.1 Background: HDTV and IP-based multimedia

High definition television (HDTV) is the next generation digital TV standard. It provides significantly higher resolution, frame rate and colour depth than standard television or teleconferencing image formats, uses a wide screen 16:9 aspect ratio, and is a purely digital media. The aspect ratio makes the image appear more "movie-like" and the greater resolution, frame rate and colour fidelity considerably enhance the realism and sense of presence inherent in a scene. Although intended for television use, the imaging formats and interconnection standards are sufficiently general purpose that cameras and other equipment can be used without modification in teleconferencing and other applications, providing significant quality improvements.

There are several HDTV imaging standards, with different resolution and frame-rate. Most commonly used are SMPTE-296M, providing a 1280x720 pixel progressive scan images at 60 frames-per-second, and SMPTE-274M, providing 1920x1080 pixel images, typically interlaced at 30 frames-per-second (a 60 frame-per-second progressive scan variant is defined, but infrequently implemented). These should be compared with the typical resolution of commercial teleconferencing systems, where CIF format images (352x288 pixels) are considered high-quality, and where the QCIF format (176x144 pixels) is still commonly used.

Interconnection of HDTV equipment is by coaxial cable according to the SMPTE-292M standard. This provides a universal medium of interchange between various types of HDTV equipment (e.g. cameras, encoders, VCRs, editing systems), and is a digital serial connection at 1.485 Gbps. It is widely used in television studios and production facilities, allowing content to be delivered uncompressed through the various cycles of capture, editing and production, avoiding the artefacts that are an inevitable result of multiple compression/decompression cycles. If wide area transport of uncompressed video is desired, the SMPTE-292M bit-stream is typically run over dedicated optical fibre connections, or using leased ATM circuits, but a more economical alternative is desirable. We consider the use of IP networks for this purpose.

Technical standards for real-time multimedia over IP are relatively mature, and systems using them have been deployed in a wide range of environments. The dominant media transfer standard is the Real-time Transport Protocol, RTP, accompanied by various profiles and payload formats that adapt it to particular application scenarios and media formats.

RTP is typically run over UDP/IP. This provides a best-effort packet delivery service, meaning that there is no guarantee that the network will not discard, duplicate, delay or reorder packets. Applications and transport protocols built IP must adapt to these issues, abstracting the network behaviour to give a usable service. RTP provides a number of services that ease this task, and applications have developed sophisticated strategies for dealing with timing jitter and packet loss. However, RTP based systems are poor at congestion control, adapting their behaviour to fit the available network capacity. The implication here is that it is necessary to either develop congestion control for RTP or to run applications only on a network provisioned with sufficient capacity to support their needs. Of course, if it is desired to transmit uncompressed HDTV over IP, the network will need a certain capacity.

A system to transport HDTV over IP networks will use RTP as its transport, with the implication being that an RTP payload format needs to be developed for HDTV content. It is expected that existing strategies for packet loss protection and timing recovery can be used, although these need to be validated in the context of high rate content. Algorithms for congestion control must be developed if it is desired to use HDTV over public IP networks, rather than those provisioned for this application.

5.2.2 System Architecture

A system for transport of HDTV over an IP network will accept a SMPTE-292M digital video signal, and encapsulate it within RTP for transmission over IP. At the receiver, the SMPTE-292M signal can be regenerated for interoperability with other equipment, or the video can be displayed or manipulated directly. There are a number of options in how this can be done, depending on the aim of the transport. If the intent is to link existing equipment the correct approach may be *circuit emulation*, where the SMPTE-292M circuit is transparently conveyed across the IP network, irrespective of its contents. The alternative is a *media aware format*, where an RTP payload format is defined to transport the image data in an optimised manner, relegating SMPTE-292M to the role of a local interconnect. There are advantages and disadvantages to each:

- A **circuit emulation** format provides transparent delivery of the HDTV bit-stream, suitable for input into other devices. It supports any format that SMPTE-292M supports, without having to be adapted to the details of that format, and accordingly is suitable for simple devices that interconnect existing equipment. The main disadvantage is that the packetization is media unaware, and cannot optimise based on the video format. This makes circuit emulation somewhat loss intolerant.

- A **media aware** format looks at the contents of the SMPTE-292M stream, acting on the video data within it. Hence, native formats may need to be defined for every possible video resolution, although those formats can be made more optimal. It also directly exposes the content to manipulation by end systems, rather than hiding it within another layer of framing.

We initially collaborated with Tektronix, Inc. to devise a circuit emulation format, that they could implement on dedicated hardware as part of their UNAS work (also supported by DARPA under the NGI programme). As advances in commodity hardware made it possible, we developed a media aware format and our own demonstrator system, to compare the two approaches. We next discuss the design of these two payload formats, followed by the implementation and performance of the demonstrator we have developed.

5.2.3 Payload Format Design

Initially, we worked with Tektronix, Inc. to design a suitable RTP payload format for use with their HDTV-over-IP implementation: the demonstrator for their Universal Network Access System (UNAS). The UNAS platform is a dedicated hardware device that provides an interface between different link technologies – in this case between IP and SMPTE-292M – and cannot process, display or otherwise manipulate the media. Accordingly, circuit emulation is appropriate: a UNAS takes the SMPTE-292M bitstream as input and encapsulates the circuit above the IP network; another UNAS receives the encapsulated bitstream, and exactly reproduces the SMPTE-292M bitstream as its output.

Circuit emulation formats map a circuit switched link onto a series of RTP packets, largely independent of the media transported on the original link. Additional payload headers are included to allow error detection and correction, sequencing and timing recovery. This allows the receiver to reconstruct the original data link format, independent of the data transported within the original link. Circuit emulation maps well onto the RTP framework, with little overhead.

Since an emulated circuit transparently delivers the underlying bitstream, it well suited to linking existing hardware devices. In particular, an emulated circuit can support any source format supported by the original network, without having to be adapted to the details of the source format. This is advantageous, since it can reduce complexity of the packetization, yet it presents a number of serious drawbacks. In particular, an emulated circuit cannot be optimised based on the video payload transported within the circuit, nor can it adapt to differing network load and congestion situations, since the original circuit is non-adaptive. Emulated circuits are also vulnerable to loss, because the original bitstream was designed to operate on a reliable link.

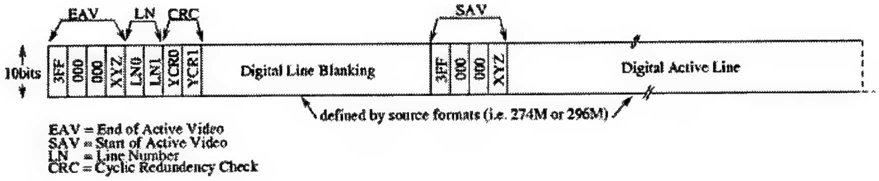

EAV
LN
CRC
SAV
10bits
3FF
000
000
XYZ
LN0
LN1
YCR0
YCR1
Digital Line Blanking
3FF
000
000
XYZ
Digital Active Line
defined by source formats (i.e. 274M or 296M)
EAV = End of Active Video
SAV = Stan of Active Video
LN = Line Number
CRC = Cyclic Redundency Check

Figure 13: The SMPTE-292M bitstream format

A SMPTE-292M circuit comprises two interleaved streams, one containing the luminance samples, the other chrominance values. Each stream is divided into four parts as illustrated in Figure 13: (1) start of active video timing reference (SAV); (2) digital active line; (3) end of active video timing reference (EAV); and (4) digital line blanking. The bitstream may also carry horizontal ancillary data (H-ANC) or vertical ancillary data (V-ANC) instead of the blanking level, and likewise, ancillary data may be transported instead of a digital active line.

The EAV and SAV are made up of three 10 bit words, with constant values of 0x3FF 0x000 0x000 and an additional word carrying a number of flags. This includes an F flag which designate which field of an interlaced frame the line is transporting and a V flag which indicates field blanking. After the EAV marker, are the line number and a Cyclic Redundancy Check. The picture data for the line follows: the number of words and format for active lines and line blanking is defined by source format documents, e.g. SMPTE 274M and 296M.

Our circuit emulation format divides each line of video in the SMPTE 292M bitstream into several RTP packets. This includes all timing signals, blanking levels, active lines and/or ancillary data. Start of active video (SAV) and end of active video (EAV+LN+CRC) signals must not be fragmented across packets, as the SMPTE 292M decoder uses them to detect the start of scan lines.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| V |P|X|   CC  |M|    PT       |      sequence# (low bits)     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                           time stamp                          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                             SSRC                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     sequence# (high bits)     |F|V| Z |       line no         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
:                        SMPTE 292M data                        :
:                                                               :
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 14: Format of the circuit emulation payload format

As shown in Figure 14, the circuit emulation format uses the standard RTP header, followed by a four octet payload specific header. The header fields are used as follows:

- The end of a video frame (the packet containing the last sample before the EAV) is marked by the M bit in the RTP header.
- The payload header contains a 16 bit extension to the standard RTP sequence number, to accommodate the high data rate of an HDTV signal. At 1.485 Gbps, with packet sizes of at least one thousand octets, a 32 bit sequence number allows for an approximate 6 hour period before wrap-around. Given the same assumptions, the standard 16 bit RTP sequence number wraps around in less than a second, which is clearly not sufficient for the purpose of detecting loss and out of order packets.
- The RTP timestamp runs at 148.5 MHz, allowing the receiver to reconstruct the timing of the SMPTE 292M stream without knowledge of the exact type of source format (e.g. SMPTE 274M or SMPTE 296M). With this timestamp, the location of the first byte of each packet can be uniquely identified in the SMPTE 292M stream. At 148.5 MHz the 32 bit timestamp wraps around in 21 seconds.
- The payload header also carries the 11 bit line number from the SMPTE 292M timing signals. This provides more information at the application level and adds a level of resiliency, in case the packet containing the EAV is lost. The F and V fields match the corresponding fields in the SMPTE-292M bitstream.

It is desirable to octet-align the picture when it is packed into RTP packets, and also adhere to the principles of application level framing. This translates into not fragmenting related luminance and chrominance values across packets, accordingly groups of 2 pixels (at 20 bits per pixel) are packed into 5-octet pixel groups, and the packet is formed from an integer number of pixel groups. The SAV and EAV+LN+CRC fields are also not fragmented across packet boundaries. Together, these rules provide some resilience to packet loss.

This payload format is fully described in the references. We have pursued standardisation of the format within the Internet Engineering Task Force, where it is currently in the process of being published as a Proposed standard RFC. It is implemented in the Tektronix UNAS prototype, and was demonstrated at the Super Computing 2001 conference: the digital output from an HDTV camera at the University of Washington was transported over an emulated circuit to the conference in Denver, with the reconstructed bitstream being passed to a standard D/A converter and display device.

In addition to the circuit emulation format, we have devised a media aware RTP payload format for uncompressed video. A media aware format uses knowledge of the video encoding to derive a concise and optimised payload format, considering application level framing. The goal is a format that it is both robust to the vagrancies of a best-effort network, adaptable to changes in network capacity, and suitable for manipulation and processing in the digital domain.

When considering uncompressed video, the structure of the frames is made visible at the RTP layer, and used to optimise the payload format. The goal is not to transparently convey the bitstream, but rather to convey the video data and associated meta data. Depending on the payload format, it may not be possible to exactly reconstruct the original bitstream.

Media aware formats use timestamps to represent the timing of the frames, rather than the bit-stream in which they are embedded. Scan line numbers and offsets are made visible. This gives the application flexibility in error recovery, since it can tell which parts of a frame are damaged, and allows use of conditional replenishment and motion vectors to reduce the data rate. Media aware formats are directly influenced by the video format, colour space, image size, etc. It is often necessary to convey parameters that describe the image to the decoder, before it can begin rendering the media stream. These parameters can be conveyed in-band or out-of-band.

An encoder for a media aware format is required to understand the media, and may be more complex than that for a circuit emulation format. The reward for this complexity is flexibility: the encoder is free to change the frame rate, image format, or encoding to match changes in network conditions and to adapt the stream to changes in network capacity. This can make media aware formats considerably more robust than circuit emulation formats which are constrained by the limitations of the circuit that must be regenerated.

Figure 15 displays the RTP payload header for our media aware payload format. Each packet contains one, or more, video scan lines. For each scan line in a packet, the standard RTP header is followed by an 8 octet payload header, indicating the scan line, offset within the scan line, and number of samples present. Other fields indicate which frame of interlaced video is represented, and if more scan lines follow within the packet.

```
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| V |P|X|   CC  |M|     PT      |       Sequence Num            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                          Time Stamp                           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                            SSRC                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     Scan Line Num             |       Scan Offset             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|          Length               |F|C|         Z                 |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     Scan Line Num             |       Scan Offset             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|          Length               |F|C|         Z                 |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
:                                                               :
:                 Two (partial) lines of video data             :
+---------------------------------------------------------------+
```

Figure 15: Media aware payload format

This format exposes the picture framing, making it straightforward to manipulate, display, or otherwise process the images. It is the basis for our implementation, described in section 5.2.4.

One could say that media aware formats take entirely the opposite approach to transport from circuit emulation. Where circuit emulation treats the bit stream as a largely opaque entity, suitable as input to a hardware device, the goal of a media aware format is to transport the media in an concise and optimised manner, taking into account the details of the video format.

Naturally, the difference between these two approaches has a number of consequences. While media aware packetization schemes are flexible and adaptable to changes in network conditions, circuit emulation is rigid. For example, when congestion occurs the only option for an emulated circuit is to terminate the connection. However, media aware transport can adapt to changes in capacity. This limits the applicability of circuit emulation to networks that are well provisioned, or allow resource reservation, and can make it unsuitable for use on the public Internet.

Media aware packetization schemes utilize a cornucopia of adaptation techniques. For example, if frames are exposed, it is possible to vary the video frame rate. Similarly, if scan line fragments are exposed, it is simple to use conditional replenishment to reduce the data rate of a signal. The data rate can also be reduced by changing the colour depth of the video stream. Circuit emulation formats, in contrast, are constrained by the format of the original bitstream. The meta data required to allow flexible transport is buried within the bitstream, and is expensive to extract, especially when operating at high rates. As a result, these formats typically operate at a single rate, with little in the way rate adaptation.

The flexibility of media aware packetization also lends itself to more robust designs. Media aware schemes expose the details of any loss, and hence allow the application flexibility in repair. For circuit emulation robustness is limited to recovery of the bitstream synchronization, and is not necessarily appropriate for timely recovery of the picture.

5.2.4 Implementation

In the design and implementation of our media-aware HDTV-over-IP prototype, our priority was to use commercial, off-the-shelf, components rather than to develop custom hardware. Accordingly, our system is built using high-performance PCs with gigabit Ethernet and HDTV capture/display cards.

Our sending and receiving hosts are Dell PowerEdge 2500 servers with dual 1.2GHz Pentium III Xeon processors, running Linux 2.4.18. They are equipped with 3Com 3c985 gigabit Ethernet network interfaces (we experimented with a range of network interfaces; with appropriate tuning all the gigabit Ethernet interfaces we tried could sustain line rate provided jumbo-frames were used). For HDTV capture and playout, we use a DVS HDstation card. This is used to capture HDTV into main memory from a SMPTE-292M link, and (optionally) to regenerate the SMPTE-292M output at the receiver. Our system can also display HDTV content on the workstation monitor, using a standard AGP graphics card. The HDstation card supports a range of video formats, but our system uses only SMPTE-296M (1280x720 pixels, progressive scan, 60 frames per second) at this time.

Key to achieving high performance at the sender is to use a system with dual PCI bus interfaces: one for video capture and one for network transmission. The receiver is similarly designed, with either dual PCI bus interfaces (if the HDstation card is used for video output) or a single PCI bus with the gigabit Ethernet card, and an AGP display card. Performance on systems with a single PCI bus is poor, due to bus contention and bandwidth limitations.

We use an updated version of the open-source RTP library from University College London to provide the core network functions of our system. This is a complete RTP implementation, including RTCP, and supports IPv4, IPv6 and multicast. We have updated this library to support the significantly higher throughput of our application, implementing zero-copy reception, optimising header validation code, tuning network parameters to increase throughput, and implementing the media-aware payload format we discussed in section 5.2.3.

Above the RTP library, we implemented transmission and reception as two separate programs running on separate PCs, because the video bandwidth is such that it is not possible to transmit and receive on the same machine (i.e. each site has two hosts; one to transmit, one to receive). The transmitter is responsible for frame capture, fragmentation to match the network MTU, packetization and transmission. Video capture is performed in a separate thread from fragmentation, packetization and transmission (because of the blocking capture API provided). The native data rate of the SMPTE-296M video signal is slightly above that of gigabit Ethernet, so the video capture hardware is programmed to perform colour sub-sampling from 10 bits per component to 8 bits per component, a data rate of 850Mbps. The colour sub-sampled signal is visually indistinguishable from the original in all but the most demanding environments; and as 10 gigabit Ethernet becomes available it can be trivially substituted to support full colour images.

The receiver is a single-threaded looping implementation, operating in a classic `select()` loop. Each iteration pulls a packet from the RTP stack, performs colour conversion if needed, and inserts the contents into a frame store at the appropriate point. If the packet is the last in the frame, rendering is triggered. Limited buffering is provided to smooth network timing jitter and ensure smooth playback. We also include a packet loss concealment algorithm, to hide the effects of any lost data. As noted in section 5.2.5, packet loss was a rare event on the NGI SuperNet test bed, and hence we did not place emphasis on robustness to loss; future systems could easily be made more robust. The receiver system also collects performance statistics and performs RTCP processing.

A key design goal of both sender and receiver is to avoid data copies, so that the system can support the required data rate. This involves scatter sends and receives (implemented using the `recvfrom()` system call with `MSG_PEEK` to read the RTP header, followed by a second call to `recvfrom()` to read the data.

Complete details of our system design and implementation are provided in the references. The software is available for download from http://www.east.isi.edu/projects/NMAA/, along with exact details of the computer, network and HDTV hardware required.

5.2.5 Performance Evaluation

Our initial trials were conducted between two hosts on the same Ethernet segment, connected by an Extreme 5i gigabit Ethernet switch. The aim was to demonstrate that our system could support HDTV delivery on an unloaded network, free from the effects of competing traffic. The tests were successful: when correctly tuned, our implementation is loss free and exhibited negligible timing variation (jitter) in the local area tests. The tuning effort was significant, however, requiring adjustments to the network MTU, socket buffer size and network driver parameters (the references describe the tuning process in detail).

We also performed a number of wide-area experiments, using the SuperNet test bed and Internet2. Our experiments were conducted on the cross-country path between ISI East in Arlington, Virginia, and the main ISI site in Los Angeles, California. The path was symmetric, with eleven hops and approximately 67ms round-trip time. Our test traffic shared the path with other IP traffic: no QoS or resource reservation was used, although the network was monitored to ensure that we did not significantly overload the links.

When the underlying network is lightly loaded, we have consistently been able to run cross-country HDTV-over-IP at 850 Mbps without packet loss. As the network becomes more loaded, typically during business hours, we occasionally see packet loss in our application, indicating small amounts of congestion in the network.

The network typically shows relatively low timing variation (jitter), as shown in Figure 16. The network preserves the relative inter-packet arrival times with remarkable accuracy; indeed our prototype implementation that presents each frame as soon as it is complete (without buffering to smooth the presentation times) provides acceptable quality.

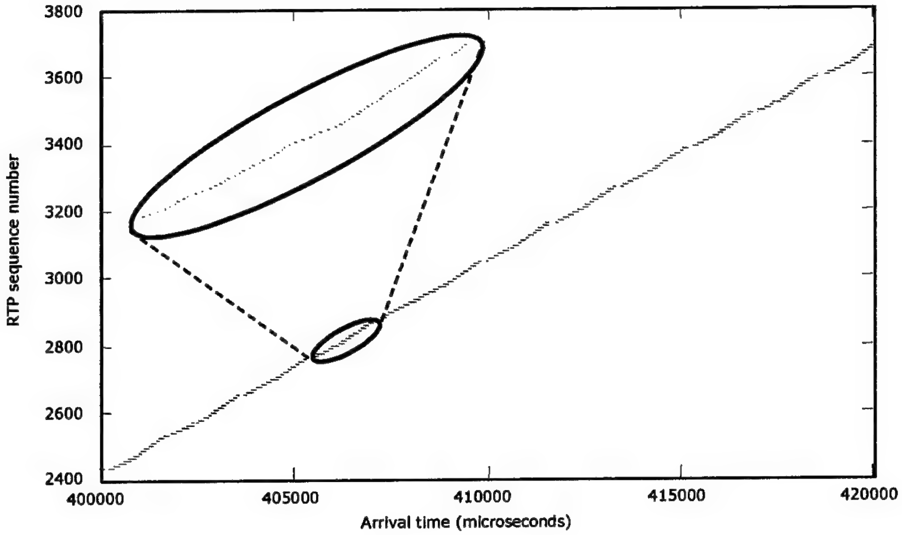


Figure 16: Inter-packet timing variation

One area where the network does exhibit surprising behaviour is in terms of packet reordering. Although inter-arrival times are typically preserved, we have observed occasional (less than 0.1%) instances where packets arrive out of sequence. Figure 17 plots the number of non-consecutive packets received, measured by non-unity sequence number increments, in a 10 million packet trace. It is clear that most out-of-order packets are the result of two consecutive packets arrived swapped in time, but instances of packets arriving 20-30 out of sequence were observed. This reordering may be caused by link-layer parallelism within the network, or by equal cost multi-path routing. We are conducting further work to characterize the scale of the problem. These out-of-order packets are not problematic for our application, but may cause issues with applications that use TCP or TCP-like congestion control.

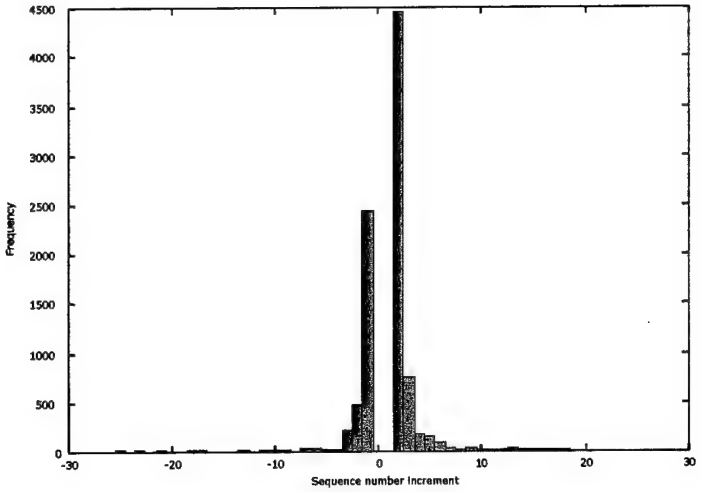


Figure 17: Packet reordering in the network

Despite the presence of occasional reordered packets, our experience with the wide area IP network has been positive: over 99.9% of packets arrived in order, with minimal jitter. The matches the results of other researchers, and clearly demonstrates the potential for IP networks to support telepresence and other high definition media. Indeed, our system is visually indistinguishable from local area HDTV transmission, and provides a remarkable degree of presence.

6 Important Findings and Conclusions

Our Digital Amphitheatre demonstrator illustrates the feasibility of conducting large scale meetings in a shared virtual environment. The Digital Amphitheatre introduced two key innovations:

- The use of synthetic background substitution to place participants into a coherent virtual space dramatically improves system usability, allowing the construction of a true networked virtual environment with a significant feeling of presence.
- The use of spatial tiling agents within the network as aggregation points to reduce the packet rate by an order of magnitude, and significantly enhance the scalability of the system.

Together, these illustrate the potential performance gains that can be achieved through novel use of standard protocols and system components. We find that many applications do not exploit the potential of the network, due to inappropriate architectures and reliance on traditional design techniques.

Our HDTV-over-IP system illustrates the feasibility of using the Next Generation Internet for high definition video teleconferencing and other applications that require high quality motion imagery. Once again, through careful design and engineering, we illustrate the capabilities of the network and its potential to support application significantly beyond those that are currently deployed.

We conclude that the Next Generation Internet has the capabilities to support the kind of demanding applications that have previously been feasible only on special-purpose networks. In particular, the NGI has been shown to be an appropriate base for large scale virtual environments, high definition teleconferencing, and other high quality motion imagery applications. This finding has the potential to be of significant benefit both for DARPA and for the community in general. Applications that have previously been considered too specialised to run on commodity network hardware – whether on the public infrastructure or on a private network leveraging COTS systems – have been demonstrated, and if developed further could be deployed to great cost savings. In addition, new applications – mass telepresence, surveillance and remote operation – are also enabled at less cost than would previously have been expected.

7 Significant Development

As described in section 5, the project has completed two demonstrator software systems: a prototype of the Digital Amphitheatre, and a demonstrator HDTV-based video teleconferencing system.

The prototype Digital Amphitheatre is described in section 5.1, and comprises a set of modifications to the video teleconferencing tool *vic*, originally from Lawrence Berkeley National Laboratory, a set of modifications to the UCL Robust-Audio Tool, *rat*, and newly developed Spatial Tiling Agents. These are as follows:

- Modifications to *vic* that enable image segmentation and synthetic background substitution.
- Modifications to *vic* that provide the Digital Amphitheatre user interface
- Modifications to *vic* that provide support for the YUVCR and tiled H.261 codecs
- Modifications to *vic* that integrate audio controls, and provide remote control of *rat*.
- Modifications to *rat* that integrate the remote control interface provided by our version of *vic*.
- Newly developed Spatial Tiling Agents, performing the image tiling function to improve the performance and scalability of the system.

The demonstrator HDTV-based video teleconferencing system is described in section 5.2, and comprises the following:

- Modifications to the University College London RTP library, for increased performance
- Newly developed code to capture HDTV content from a SMPTE-292M format source; and to transmit that content over IP networks at gigabit rate, using the RTP payload format we have designed.
- Newly developed code to receive transmissions of HDTV content over IP networks at gigabit rate, using the RTP payload format we have designed; to recover the timing of the resulting signal; to conceal the effects of network packet loss; and to regenerate the SMPTE-292M signal suitable for display or interconnection with other equipment.

Source code for both systems is available for download from http://www.east.isi.edu/projects/NMAA, under a “BSD-style” open source license. The source code has been tested on Red Hat Linux 7. Documentation and complete descriptions of the system hardware requirements are also provided.

8 Implications for Further Research

Our Digital Amphitheatre system has demonstrated the feasibility of conducting large scale distributed meetings using the Next Generation Internet infrastructure. It is a proof-of-concept system, in need of significant engineering development before it can be widely deployed. In terms of research, it clearly shows the benefit of distributing processing throughout the network, when it is necessary to scale a system to support large numbers of simultaneous participants, distributed across the wide area. It should, however, be noted that our design is based on active *agents*, rather than an *active network*. It is clear, from our research, that the network infrastructure developed under the NGI programme is sufficient to support this class of massively distributed application; provided that it can efficiently support a location service. This leads to the following recommendations:

- Large scale distributed applications of this type can benefit from an efficient naming and location service, to find and contact agents and service providers within the network. This may require advances in network routing infrastructure, or small additions of functionality, but it does not require a fully active network.
- The Digital Amphitheatre user interface illustrates the importance of usability in making a system suitable for purpose. Distributed user interface processing, and seamless integration of real-world participants with the virtual environment are clearly areas where productive research can be conducted.

Our HDTV-over-IP system has demonstrated the feasibility of very high quality video teleconferencing and other motion imagery distribution using the NGI network. It has been tested across a number of real-world networks, and is robust and suitable for widespread demonstration and testing. The system shows clear benefits over special-purpose networks, but still has a number of limitations. We consider the following future research issues:

- The Next Generation Internet programme has excelled in improving network capacity, to the extent where it is possible to conduct gigabit rate video teleconferences on a best effort network. If such applications are to be routinely deployed, it will be necessary to implement congestion control and/or quality of service functions, to provide smooth operation in times of network overload. Neither of these are well understood problems, in the context of an IP network.
- Our demonstrator operates at the limit of end-system performance. Whilst it may be expected that Moore's law will solve this, as it has many other performance problems, this is not the case. The bottlenecks in our system are due to I/O performance and operating system issues, not limitations of the CPU. We urge development of the PC platform to improve the sustained throughput, and simultaneous transmit/receive throughput. We also urge development of the operating system, to make effective use of the hardware.

In summary, the proof-of-concept is available, but there is a need for further research into routing and naming, congestion control and quality of service.

9 Bibliography

Copies of the conference and journal publications, standards contributions and presentations referenced here are available online at http://www.east.isi.edu/projects/NMAA/.

9.1 Conference and Journal Publications

- Ladan Gharai, Colin Perkins, Ron Riley and Allison Mankin, *Large Scale Video Conferencing: A Digital Amphitheatre*, Proceedings of the 8th International Conference on Distributed Multimedia Systems, San Francisco, CA, USA, September 2002.
- Ladan Gharai and Colin Perkins, *Implementing Congestion Control in the Real World*, Proceedings of the IEEE International Conference on Multimedia and Expo, Lausanne, Switzerland, August 2002.
- Ladan Gharai, Colin Perkins and Allison Mankin, *Large Group Teleconferencing: Techniques and Considerations*, Proceedings of the 3rd International Conference on Internet Computing, Las Vegas, NV, USA, June 2002.
- Colin Perkins, Ladan Gharai, Tom Lehman and Allison Mankin, *Experiments with Delivery of HDTV over IP Networks*, Proceedings of the 12th International Packet Video Workshop, Pittsburgh, PA, USA, April 2002.
- Ladan Gharai, Colin Perkins and Allison Mankin, *Scaling Video Conferencing through Spatial Tiling*, Proceedings of the 11th International Workshop on Network and Operating Systems Support for Digital Audio and Video, Port Jefferson, NY, USA, June 2001.
- Colin Perkins and Allison Mankin, *Diversinet: Embracing heterogeneity in future network services*, Workshop on New Visions for Large-Scale Networks: Research and Applications, Vienna, VA, USA, March 2001.

- Allison Mankin, Ladan Gharai, Ron Riley, Maryann Perez Maher and Jaroslav Flidr, *The Design of a Digital Amphitheatre*, Proceedings of the 10th International Workshop on Network and Operating System Support for Digital Audio and Video, Chapel Hill, NC, USA, June 2000.

9.2 Standards Contributions

- Ladan Gharai, Colin Perkins, Gary Goncher & Allison Mankin, *RTP Payload Format for SMPTE 292M Video*, Internet Engineering Task Force, Approved for RFC publication, January 2003.
- Ladan Gharai and Colin Perkins, *RTP Payload Format for Uncompressed Video*, Internet Engineering Task Force, Work in Progress, November 2002.

9.3 Presentations and Demonstrations

- Colin Perkins and Tom Lehman, *High Performance Networks and Multimedia*, Presentation and demonstration for KidzOnline, Arlington, VA, USA, December 2002.
- Ladan Gharai, *RTP Payload Format for Uncompressed Video*, Presentation and discussion in the Audio/Video Transport working group at the 55th Internet Engineering Task Force meeting, Atlanta, GA, USA, November 2002.
- Colin Perkins, Alec Aakesson and Tom Lehman, *HDTV over IP*, Demonstration at the Super Computing 2002 conference, Baltimore, MD, USA, November 2002.
- Colin Perkins and Tom Lehman, *HDTV over IP*, Presentation and demonstration to engineers from PBS, Arlington, VA, USA, November, 2002.
- Ladan Gharai, *Uncompressed high quality video over IP*, Presentation and discussion in the Audio/Video Transport Working Group at the 54th Internet Engineering Task Force meeting, Yokohama, Japan, July 2002.
- Colin Perkins, *RTP: Multimedia Streaming over IP*, Presentation at the Video Services Forum, Austin, TX, USA, June 2002.
- Ladan Gharai and Colin Perkins, *Scaling Multimedia Conferencing*, Presentation to Microsoft Research, Redmond, WA, USA, June 2002.
- Colin Perkins, *Experiments with Delivery of HDTV over IP Networks*, Presentation at the 12th International Packet Video Workshop, Pittsburgh, PA, USA, April 2002.
- Ladan Gharai, *RTP Payload Format for SMPTE292M*, Presentation and discussion in the Audio/Video Transport working group of the 53rd Internet Engineering Task Force meeting, Minneapolis, MN, USA, March 2002.
- Colin Perkins, *NGI Multicast Applications and Architecture*, Presentation and demonstration to DARPA NGI PI Meeting, Tyson's Corner, VA, USA, January 2002.
- Colin Perkins, *Real-Time Delivery of Motion Imagery over IP Networks*, Presentation at the NSF/NIMA workshop on Defining a Motion Imagery Research and Development Program, Herndon, VA, USA, November 2001.
- Colin Perkins and Allison Mankin, *HDTV over IP*, Demonstration at the Super Computing 2001 conference, Denver, CO, USA, November 2001.
- Ladan Gharai, *Scaling Video Conferencing Through Spatial Tiling*, Presentation at the 11th International Workshop on Network and Operating Systems Support for Digital Audio and Video, Port Jefferson, NY, USA, June 2001.
- Colin Perkins and Gary Goncher, *Uncompressed HDTV over IP*, Presentation at the DARPA workshop on Internet HDTV, Seattle, WA, January 2001.
- Ladan Gharai, *RTP Payload Format for SMPTE292M*, Presentation and discussion in the Audio/Video Transport working group at the 49th Internet Engineering Task Force meeting, San Diego, CA, USA, December 2000.
- Ladan Gharai, Colin Perkins and Allison Mankin, *Digital Amphitheatre*, Demonstration at the Super Computing 2000 conference, Dallas, TX, USA, November 2000.

- Allison Mankin and Colin Perkins, *Digital Amphitheatre*, Presentation and Demonstration at the Ubiquitous Computing Workshop of the DARPA Expeditions Program, Pittsburgh, PA, USA, October 2000.
- Colin Perkins, *NGI Multicast Applications and Architecture*, Presentation to DARPA NGI PI Meeting, McLean, VA, USA, October 2000.
- Ladan Gharai, *RTP Payload Format for SMPTE292M,* Presentation and discussion to the Audio/Video Transport working group at the 48th Internet Engineering Task Force meeting, Pittsburgh, PA, USA, August 2000.
- Ron Riley, *The Design of a Digital Amphitheatre*, Presentation at the 10th International Workshop on Network and Operating System Support for Digital Audio and Video, Chapel Hill, NC, USA, June 2000.